APPLICATION FOR UNITED STATES LETTERS PATENT

by

J. THOMAS NGO

STEVEN E. SAUNDERS

and

OWEN RUBIN

for

SYSTEM AND METHOD FOR THE BROADCAST DISSEMINATION OF TIME-ORDERED DATA WITH MINIMAL COMMENCEMENT DELAYS

Burns, Doane, Swecker & Mathis, LLP
Post Office Box 1404
Alexandria, Virginia 22313-1404
(703) 836-6620

Attorney Docket No. 032516-001

SYSTEM AND METHOD FOR THE BROADCAST DISSEMINATION OF TIME-ORDERED DATA WITH MINIMAL COMMENCEMENT DELAYS

Field of the Invention

The present invention is directed to systems for broadcasting time-ordered data, such as streaming media presentations, and more particularly to a system which minimizes the delay between a user's indication of a desire to receive the data and the commencement of the utilization of the data, with efficient use of the broadcast bandwidth.

Background of the Invention

A variety of situations exist in which users desire access to time-ordered data, and prefer to begin utilizing that data as soon as possible after they have entered a request for the data. One example is the reproduction of streaming media data, such as a video presentation or a musical rendition. In the case of a video, for instance, a television system subscriber may desire to view a particular movie that is being broadcast by a cable or satellite television network.

In a conventional television system, the subscriber is required to wait until the scheduled time at which the movie is to be broadcast before it is possible to begin viewing it. For instance, a given movie may be repetitively broadcast on a particular channel every two hours. If the subscriber becomes aware of the movie's availability one half hour after the most recent broadcast has begun, it is necessary to wait an hour and a half before the movie can be viewed in its entirety from start to finish. As an alternative, the subscriber could choose to ignore the first half hour that was missed, and view the remaining portion of the movie. In many situations, however, this alternative is unacceptable, particularly where the initial portion of the movie is essential to an understanding of events that occur later in the movie.

To alleviate this situation, different approaches have been employed to reduce the time that a subscriber must wait until the next instance at which the

beginning of the movie becomes available. In general, the maximum potential waiting period is reduced by allocating a greater amount of bandwidth, e.g., number of channels, to the movie. In certain situations, the available bandwidth is sufficient to provide true video-on-demand to all viewers, whereby the movie begins to be transmitted in its entirety to any viewer immediately upon receiving a request for the movie. In practice, however, this capability can only be cost-effective in a relatively small, closed environment, such as within a hotel or an apartment building. In these situations, where the number of viewers at any given time is limited, it is possible to effectively dedicate a channel to each viewer and to begin a transmission of the movie over a viewer's channel upon receipt of a request to begin the movie.

This approach to video-on-demand services is not economically feasible in a broadcast environment, where the number of potential viewers is so large that the required bandwidth becomes cost-prohibitive. Consequently, broadcasters employ an approach that is labeled "near-video-on-demand". In this approach, a popular movie is broadcast at multiple staggered start times over a limited number of channels. For instance, a two-hour movie can be broadcast over four channels at 30 minute intervals. Consequently, the viewer never has to wait more than 30 minutes for the start of the next presentation. With this approach, the number of channels occupied by a given movie is inversely proportional to the wait period. If 5 minutes is considered to be an acceptable wait time, a two-hour movie would require 24 channels. A one-minute wait time would occupy 120 channels.

In an effort to simulate true video-on-demand capabilities in a near-video-on-demand environment, a class of techniques has been developed which rely upon preliminary storage of part of the movie at the viewer's premises. One example of this technique is described in U.S. Patent No. 5,884,141. In this technique, a desired movie is transmitted over multiple channels at staggered times, as in near-video-on demand. If the staggered presentation is to begin every 30 minutes, for example, the first 30 minutes of the movie, known as the leader, is automatically

stored at the subscriber's premises. When a subscriber enters a request to view the movie, for example by pressing a "play" button on a set-top converter or a remote control unit, or by dialing a specified phone number, the first 30 minutes of the movie is replayed from the locally stored leader. During this time, the remaining portion of the movie, which occurs after the leader, begins to be stored from one of the channels allocated to the movie, and is replayed in a time-shifted fashion after the replay of the stored leader has completed.

Using this approach, the number of movies that can be viewed with substantially no wait time is determined by the amount of local storage capacity at the subscriber's premises. For example, several gigabytes of storage in the set-top converter unit are required to cache a 30-minute leader for each of ten select movies. If the subscriber desires to view any movie other than the ten whose leaders have been stored, the normal near-video-on-demand wait time would be encountered, e.g., 30 minutes.

While this approach can significantly reduce viewer wait times, relative to conventional and near-video-on-demand transmissions, it is highly dependent upon the amount of storage capacity that is available at the user's premises. It is desirable to provide video-on-demand capabilities in a broadcast environment with significantly reduced storage requirements at the subscriber's premises, and without a subscriber-hardware-based limit on the number of broadcast presentations that can be viewed with minimal wait times.

A broadcast technique which offers such a possibility is described in U.S. Patent Nos. 5,751,336 and 5,936,659, as well as related publications "Pyramid Broadcasting for Video On Demand Service," by S. Viswanathan and T. Imielinski, *SPIE Proceedings*, Vol. 2417, pp. 66-78, February 1995, and "Metropolitan Area Video-On-Demand Service Using Pyramid Broadcasting" by Viswanathan and Imielinski, *Multimedia Systems*, Vol. 4, pp. 197-208, 1996. In the technique described in these documents, each movie is divided into segments of increasing size, and each segment is repeatedly transmitted in an associated

logical channel of the broadcast medium. In the disclosed implementations, each segment is approximately 2-2.5 times longer than the preceding segment. At the receiving end, once the first segment is received, all of the other segments are received in time, so that continuous viewing of the movie is possible.

While this technique provides improved results relative to the approaches described previously, the particular embodiments disclosed in the patents and related publications still require a significant amount of bandwidth. It is an objective of the present invention to improve upon the basic principles described in these references, and provide more efficient bandwidth utilization while minimizing viewer access time. It is a further objective to address some of the practical problems associated with the implementation of this technique which are not addressed in the references.

Summary of the Invention

In accordance with the present invention, the foregoing objectives are achieved by dividing a stream of time-ordered data into multiple fragments of equal length, and repetitively transmitting the fragments at different respective repetition rates. The fragments are reordered for transmission so that those which occur near the beginning of the original data stream are transmitted more frequently than those which occur later in the data stream. When a subscriber enters a request to utilize the data, e.g., view a movie, the individual fragments are stored upon receipt at the subscriber's premises, and reassembled into a contiguous stream for the subscriber. The ordering of the fragments is such that the wait time required before utilization of the data can begin is limited to a predetermined maximum, and at least one copy of every fragment becomes available by the time it is needed.

Various techniques are provided by the present invention that offer practical advantages for the broadcaster and the subscriber within this general mode of operation. In one approach, a conventionally transmitted data stream, in

which each of the fragments appear in their sequential order, accompanies the reordered transmission. This sequentially ordered stream of data can be viewed in a conventional manner by subscribers who do not have the local storage required to decode the reordered transmission. Some of the fragments that appear within the conventionally transmitted data stream are employed as substitutes, or proxies, for some of the reordered fragments. As a result, fragments can be deleted from the reordered transmission, to make the overall transmission periodic and thereby allow a given program to be broadcast indefinitely by repeatedly transmitting a finite data stream. In a further implementation, this approach can be used to reduce the time that users must wait between the end of the availability of one movie and the time at which the viewing of a new movie can begin.

Further features of the invention relate to encoding techniques that provide for effective use of the available bandwidth, and an approach to multiplexing which accommodates variable data rates within a fixed bandwidth transmission. These and other features of the invention, as well as the advantages provided thereby, are explained in greater detail hereinafter with reference to exemplary embodiments of the invention depicted in the accompanying drawings.

Brief Description of the Drawings

Figure 1 is a general block diagram of a broadcast system of the type in which the present invention can be implemented;

Figure 2 is a time line depicting a sequence of fragments in a presentation;

Figure 3 illustrates one example of the manner in which a presentation can be divided into segments for allocation to transmission substreams;

Figure 4 is a plot of the separation of fragment copies versus playback time for the embodiment of Figure 3;

Figure 5 is a plot of optimal and quantized fragment densities;

Figure 6 is an enlarged view of a portion of the plot of Figure 5, illustrating a differential change in the length of a presentation segment;

Figure 7 is a block diagram of a first embodiment of a decoder for processing data that is transmitted in accordance with the present invention;

Figure 8 is an illustration of a format for transmitting one fragment of data;

Figure 9 is a block diagram of a second embodiment of the decoder;

Figures 10a-10c illustrate the concepts which underlie the deletion of fragments from substreams and the use of proxy fragments;

Figure 11 illustrates the relationship of encoded substreams to a conventional stream;

Figure 12 illustrates the alignment of deleted fragments relative to proxy fragments in a conventional stream;

Figure 13 illustrates the switching of encoded substreams from one presentation to another;

Figure 14 is an alternative illustration of the switching of encoded substreams from one presentation to another;

Figure 15 is an illustration of a first embodiment of switching which employs reordered fragments, without a conventional layer;

Figure 16 illustrates a second embodiment of switching with reordered fragments, which includes a non-contiguous conventional layer;

Figure 17 illustrates another embodiment of switching with reordered fragments, which employs a contiguous conventional layer;

Figure 18 is a time diagram illustrating the relationship between the receipt and play of fragments;

Figure 19 is an illustration of switching with a conventional layer, using irregular stop and start boundaries;

Figure 20 is an illustration of the assignment of nominal times to fragments; and

Figure 21 is a graph depicting the results obtained by different broadcast techniques.

Detailed Description

Generally speaking, the present invention comprises a set of techniques for the efficient operation of a system which provides virtual on-demand access to temporally-ordered data that is disseminated via a broadcast medium. To facilitate an understanding of the principles which underlie the invention, it is described hereinafter with reference to a specific application thereof. In particular, reference is made to the use of the invention in the context of streaming media presentations, such as televised movies. It will be appreciated, however, that the practical applications of the invention are not limited to these particular examples. Rather, the invention can be employed in any situation in which it is desirable to provide temporally-ordered data in a manner which permits a user to enter a request to receive the data at any arbitrary point in time, and to begin utilization of the data with minimal delay after such a request.

In the context of the present invention, the term "temporally-ordered data" refers to any collection of data in which some portion of the data must be received prior to the time that another portion of the data can be utilized. For instance, in the case of a video presentation, the frames of a movie can be received in any order, and stored for subsequent presentation. However, the viewer cannot begin to watch the movie until the first frame has been received. The invention guarantees that the first frame is available with a minimal time delay, and each succeeding frame is also available by the time it is needed to recreate the original sequence. In another application, the invention can be employed to broadcast software programs, and similar types of data, in which some portions of the program are required for the user to begin its operation, whereas other portions may not be required until later in the operation, and can therefore be received after the operation has been initiated. In yet another application, the invention can be employed to broadcast media objects such as audio, video or animation in the context of a multimedia presentation.

The basic objective of the present invention is to reduce the wait time that is experienced by users between the entry of a request to receive data, such as a movie, and the time when the utilization of the data in its time-ordered fashion can begin, while minimizing the amount of bandwidth and local data storage that is needed to support such a capability. The techniques of the invention are implemented in a system in which the temporally-ordered data is divided into a sequence of small equally-sized fragments, and the fragments are repetitively transmitted in one or more streams over a suitable broadcast medium, such that the fragments at the beginning of the presentation are transmitted at a relatively high density, whereas those near the end of the presentation are transmitted with a lower density. To facilitate an understanding of the principles which underlie the invention, the basic concept of such a system is first described with reference to relatively simple examples, followed by more detailed discussions of various optimizations that are provided by the invention. In the following discussion, a stream of data which is transmitted in accordance with the foregoing principle, where fragments are re-ordered with differing densities, is identified as an "encoded" stream, whereas a stream of data in which the fragments are transmitted in their original sequential order is identified as a "conventional" stream.

General Concept

Referring to Figure 1, an encoded stream of data fragments which constitute a video presentation is transmitted from a source 10, such as a cable head-end transmission station, to multiple subscribers' premises. At each subscriber's premises, the stream of fragments is received in a suitable set-top converter 12, or other equivalent type of equipment for receiving signals from the source. A decoder 14 within the converter reassembles the fragments into a continuous video stream, and stores them in a suitable frame buffer 16, where they are sequentially presented to the subscriber's television receiver 18. The ordering of the fragments in the encoded stream is such that, regardless of any arbitrary

point in time at which any subscriber enters a request to view the presentation, the first fragment of the presentation is available within a maximum period of time τ, and at least one copy of any given fragment becomes available by the time it is needed for viewing in the proper order.

Figure 2 is a time line which illustrates the relationship of various factors that are employed throughout the following discussion. Time 0 is considered to be the moment at which the subscriber presses a "play" button or performs an analogous action to enter a request to view a presentation or otherwise utilize temporally-ordered data. Time instant t(0) is the moment at which the display of the first fragment in the media presentation begins, τ seconds later. Each fragment n of the presentation is displayed at a corresponding time t(n). The following variables are also employed in the discussion of the principles which underlie the invention:

G	Size of a fragment
n_{max}	Index of the last fragment in the presentation
T	Total running time of the presentation
v(n)	Fragment density, i.e., the number of times the n-th fragment appears in a data stream of length T
η	A bandwidth multiplier
m(n)	Nominal bandwidth of the presentation at fragment n.

The bandwidth multiplier, or expansion coefficient, η describes the amount of bandwidth that is needed to provide a wait time τ, compared to the amount of bandwidth B that would be required for a conventional data stream. For instance, a movie might be transmitted at a nominal bit rate of 3Mb/s. Thus, a bandwidth multiplier of 9 indicates a bit rate of 27Mb/s. The bandwidth multiplier can be expressed as a function of T and τ. To simplify the initial discussion of the principles which underlie the invention, it is assumed that the repetitively transmitted copies of a given fragment are uniformly distributed in time. Furthermore, it will be assumed that G is a constant, i.e. all of the fragments are

of equal size. In the case of a media presentation, the size of a fragment can be thought of in terms of a time increment of the presentation, e.g. 0.5 second. When considering transmission and storage issues, however, the size of a fragment is preferably expressed as an amount of data. In one embodiment, each fragment contains 188 bytes of data.

Each fragment n is transmitted repeatedly, ideally at time intervals of length $(T/\nu(n))$. Consequently, the first occurrence of a fragment n is no later than $(T/\nu(n))$. To guarantee contiguous assembly of the fragments at the viewer's premises, the fragment must be received no later than t(n). Therefore, the fragment density must conform to the following relationship:

$$(T/\nu(n)) \leq t(n) \tag{1}$$

or

$$\nu(n) \geq T/t(n) \tag{2}$$

Thus, the fragment density, or repetition rate, is a decreasing function of n, the fragment's location within the original presentation.

A lower bound on the bandwidth multiplier is given by the mean fragment density:

$$\eta \geq \frac{1}{T}\int_0^T \left(\frac{T}{t+\tau}\right) dt \tag{3}$$

$$= \ln\left(1+\frac{T}{\tau}\right) \tag{4}$$

$$\approx \ln\left(\frac{T}{\tau}\right) \tag{5}$$

Encoding

The fragment stream is encoded so that the individual fragments repeat with a density that satisfies the relationship defined at (2) above. A variety of different encodings are possible which comply with this relationship. In one embodiment, the presentation is divided into a plurality of segments, and each

segment is repetitively transmitted as a substream of fragments. Each of the individual substreams can be transmitted in parallel with all of the other substreams. More preferably, however, all of the substreams are transmitted in a time-division multiplexed manner, to form a complete data stream of encoded data.

The first substream contains a repeating loop of the lowest-numbered fragments in a presentation, e.g., the earliest frames in a movie. The maximum length of that segment is defined as the maximum length of time by which successive copies of the same fragment can be separated and still meet the constraints of relationship (2). As many fragments as possible are contained in the segment, until that limit is reached. The next succeeding substream contains a repeating loop of the lowest-numbered fragments that are not in the preceding substream. This segment can be at least as long as the preceding segment, and typically will be longer because the maximum spacing between its lowest-numbered fragment will be greater. Successive segments are determined in this manner, until the entire presentation is encoded. The number of substreams, which corresponds to the number of segments, determines the bandwidth multiplier η.

This encoding scheme will first be explained with reference to an example that is orderly, and which corresponds to the embodiments disclosed in the previously cited references. The first fragment in a segment i is labeled with the index n_i, where $n(0)=0$. There are a total of N segments in the presentation. In this first example, each substream i is allocated the same amount of bandwidth as that which would normally be allocated to the presentation when it is transmitted in a conventional, i.e., time-ordered sequential, manner. Typically, this bandwidth is about 3Mb/s for a video presentation that is compressed according to the MPEG encoding standard.

Figure 3 illustrates the manner in which the segments are derived from the original presentation in this initial example, for allocation to respective

substreams. The presentation is divided into a sequence of n_{max} fragments. Since the maximum wait time is defined as τ, the beginning of the presentation must be available at least every τ seconds. Therefore, the initial segment contains the first τ seconds of the presentation, in this case fragments 0 through n_1-1. The first substream, i.e., Substream 0, consists of a repeating copy of this segment.

The beginning of the remainder of the movie must be available at least every 2τ seconds, i.e., the initial wait period τ plus the length of the preceding segment, which is equal to τ. Accordingly, the second segment contains the next 2τ seconds of the movie, in this case fragments n_1 through n_2-1. The second substream comprises a repeating copy of this second segment.

The beginning of the next segment, which is 3τ seconds into the presentation, must be available every 4τ seconds, i.e., the length of the wait time τ, the first segment τ and the second segment 2τ. Therefore, the next segment contains the next 4τ seconds of the presentation, which is repeatedly transmitted in the third substream. Continuing on in this manner, each successive segment is twice as long as the previous segment, and is therefore transmitted in its respective substream at one-half the repetition rate of the previous segment.

After N segments have been obtained, the total length of the presentation is $(2^N-1)\tau$. For instance, by using seven substreams is possible to transmit a movie of length 127τ. Referring to Equation 5, the optimum bandwidth multiplier η for a movie of length 127τ is ln(127), or 4.84. Hence, an encoding scheme of the type described above, which requires seven substreams, and hence where $\eta=7$, uses more bandwidth than that which is optimal.

Figure 4 illustrates a plot of the maximum separation between copies of a fragment versus the time at which that fragment is expected to be presented. The solid line illustrates the encoding example described above, and the dashed line depicts the theoretical optimum defined by Equation 5. This plot reveals that excess bandwidth is consumed by portions of the presentation that are repeated more often than necessary. For instance, a fragment of the presentation which

occurs just prior to 3τ seconds into the presentation can, in theory, be available about every 4τ seconds. However, in the foregoing encoding scheme, it is repeated nearly every 2τ seconds. Excessive bandwidth requirements occur when a segment is too long, since the start and end of the segment can have vastly different requirements with regard to the maximum separation of successive copies of a fragment that is needed to satisfy relationship (2).

In the orderly embodiment described above, that separation varies by a factor of two within a given segment. More optimum encoding schemes can reduce that variation, through the use of shorter segments, which in turn reduces the bandwidth that is allocated to each substream. For instance, each substream could be allocated one-tenth of the bandwidth that is allocated to a conventional video signal stream, e.g., 0.3Mb/s. In this case, therefore, the first substream requires only enough bandwidth to repetitively transmit the first 0.1τ seconds of the presentation. The beginning of the remainder of the presentation must be available every 1.1τ seconds, i.e., the length of the wait time τ plus the length of the first segment. Therefore, the second segment comprises the next 0.11τ seconds of the presentation. The beginning of the third segment must be available every 1.21τ seconds, so this segment comprises the next 0.121τ seconds of the movie.

In this approach, each successive segment is 10% longer than its predecessor. After N segments have been obtained, the total length of the presentation, T, can be defined as:

$$T = 0.1\tau \sum_{i=0}^{N-1} (1.1)^i \tag{6}$$

$$= 0.1\tau \left(\frac{(1.1)^N - 1}{0.1} \right) \tag{7}$$

$$= \left((1.1)^N - 1\right)\tau \tag{8}$$

By means of this approach, a presentation of length 128τ can be transmitted in 51 substreams. Since each substream is 1/10 the bandwidth of a conventional video stream, the bandwidth multiplier η equals 5.1. It can be seen that this encoding scheme is substantially closer to the optimal bandwidth of Equation 5 than the orderly example described previously, where $\eta=7$.

The foregoing approach can be generalized by allocating each substream an amount of bandwidth λB, where B is the bandwidth of a conventional video stream, e.g., 3Mb/s, and λ is a positive number in the range from 0 to 1. In a preferred embodiment of the invention, λ is the same for every substream, i.e. each substream receives equal bandwidth. In this case, the total length of the presentation is defined by the following power series:

$$T = \lambda\tau \sum_{i=0}^{N-1} (1+\lambda)^i \tag{9}$$

$$= \lambda\tau \left(\frac{(1+\lambda)^N - 1}{\lambda} \right) \tag{10}$$

$$= \left((1+\lambda)^N - 1\right)\tau \tag{11}$$

and the bandwidth multiplier η is:

$$\eta = \lambda N \tag{12}$$

$$= \frac{\lambda}{\ln(1+\lambda)} \ln\left(\frac{T}{\tau}+1\right) \qquad (13)$$

The factor $\ln(T/\tau+1)$ is equal to the optimal bandwidth multiplier set forth in Equation 4. The excessive bandwidth is therefore represented by the factor $\lambda/\ln(1+\lambda)$, where a value of unity corresponds to optimal. Hence the excess bandwidth requirements shrink as λ is reduced toward zero, i.e. decreasing λ reduces the required bandwidth. Preferably, λ is less than or equal to 1/2, and in practical embodiments of the invention $1/3 \geq \lambda \geq 1/25$.

The foregoing analysis assumes that each segment *i* has an optimal length corresponding to a repetition period of:

$$p(i)=\tau(1+\lambda)^{i}.$$

In practice, however, it may not be possible to achieve such an optimum value for each segment. In particular, if the substreams are to be transmitted using a simple form of time-division multiplexing, each segment must contain an integral number of equal-length fragments. This requirement imposes a quantization constraint on the lengths of the segments. In effect, this means that a quantity δ must be subtracted from the optimal length of the segment. The value of δ can be up to the length of one fragment.

The foregoing analysis will now be reviewed, taking this quantization constraint into account. Since the first substream is allocated enough bandwidth to repetitively transmit the first $\lambda\tau$ seconds of the presentation, the first segment has a length of $\lambda\tau$-δ seconds. The beginning of the remainder of the presentation must be available every $\tau+(\lambda\tau-\delta)$ seconds, which turns out to be $(1+\lambda)\tau-\delta$ seconds. Therefore, the second substream can repetitively transmit the next $\lambda(1+\lambda)\tau-\lambda\delta$ seconds of the movie. Upon quantization, therefore, the second segment has a length of $\lambda(1+\lambda)\tau-(1+\lambda)\delta$ seconds. Continuing on in this fashion, it will be seen

that the i-th substream repetitively transmits $(\lambda\tau-\delta)(1+\lambda)^i$ seconds of the presentation.

When the quantization factor is considered, the total length of the presentation becomes

$$T = (\lambda\tau-\delta)\sum_{i=0}^{N-1}(1+\lambda)^i \tag{14}$$

$$= \tau' \left[(1+\lambda)^N-1\right] \tag{15}$$

$$= (\lambda\tau-\delta)\left[\frac{(1+\lambda)^N-1}{\lambda}\right] \tag{16}$$

where

$$\tau'=\tau-\frac{\delta}{\lambda} \tag{17}$$

The following Table 1 illustrates a sample encoding for a 2-hour presentation, such as a movie, in accordance with the foregoing concepts. In this example, each fragment consists of 0.5 second, e.g. 1.5Mb, and the parameter λ equals 1/3, so that each substream is allocated approximately 1Mb/s. The value of τ is 27 fragments, or 13.5 seconds, and the quantization factor δ is 1.5 seconds, to provide a maximum wait time of 15 seconds. Since there are 22 substreams, the aggregate bandwidth requirement is 22Mb/s, so that the bandwidth multiplier $\eta=7.33$.

Each row of Table 1 corresponds to one substream. Hence, Substream 0 consists of a repeating sequence of fragments 0 through 8, Substream 1 consists of a repeating sequence of fragments 9 through 20, etc. Each column of Table 1 represents one time slot for the time-division multiplexed transmission of all of the substreams. Thus, in the first time slot, fragments 0, 9, 21, 37, 58... are transmitted, and in the next time slot fragments 1, 10, 22, 38, 59... are transmitted. In practice, the rows of the table continue indefinitely to the right, until such time as the transmission of the presentation is terminated. The encoded presentation is stored at the source 10, and repeatedly transmitted over the substreams, as shown in Table 1.

Sub-stream	Fragment Sequence																				
0	0	1	2	3	4	5	6	7	8	0	1	2	3	4	5	6	7	8	0	1	•••
1	9	10	11	12	13	14	15	16	17	18	19	20	9	10	11	12	13	14	15	16	•••
2	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	21	22	23	24	•••
3	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	•••
4	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	•••
5	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100	101	102	103	104	105	•••
6	123	124	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	•••
7	173	174	175	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	192	•••
8	239	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	256	257	258	•••
9	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	342	343	344	345	346	•••
10	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463	464	•••
11	602	603	604	605	606	607	608	609	610	611	612	613	614	615	616	617	618	619	620	621	•••
12	811	812	813	814	815	816	817	818	819	820	821	822	823	824	825	826	827	828	829	830	•••
13	1090	1091	1092	1092	1094	1095	1096	1097	1098	1099	1100	1101	1102	1103	1104	1105	1106	1107	1108	1109	•••
14	1462	1463	1464	1465	1466	1467	1468	1469	1470	1471	1472	1473	1474	1475	1476	1477	1478	1479	1480	1481	•••
15	1958	1959	1960	1961	1962	1963	1964	1965	1966	1967	1968	1969	1970	1971	1972	1973	1974	1975	1976	1977	•••
16	2619	2620	2621	2622	2623	2624	2625	2626	2627	2628	2629	2630	2631	2632	2633	2634	2635	2636	2637	2638	•••
17	3501	3502	3503	3504	3505	3506	3507	3508	3509	3510	3511	3512	3513	3514	3515	3516	3517	3518	3519	3520	•••
18	4677	4678	4679	4680	4681	4682	4683	4684	4685	4686	4687	4688	4689	4690	4691	4692	4693	4694	4695	4696	•••
19	6245	6246	6247	6248	6249	6250	6251	6252	6253	6254	6255	6256	6257	6258	6259	6260	6261	6262	6263	6264	•••
20	8335	8336	8337	8338	8339	8340	8341	8342	8343	8344	8345	8346	8347	8348	8349	8350	8351	8352	8353	8354	•••
21	11,122	11,123	11,124	11,125	11,126	11,127	11,128	11,129	11,130	11,131	11,132	11,133	11,134	11,135	11,136	11,137	11,138	11,139	11,140	11,141	•••

As noted previously, the quantization of the segments to contain an integral number of fragments results in excess bandwidth requirements beyond the theoretical optimum set forth in Equation 4. Figure 5 illustrates the fragment density $\nu(n)$ over the length of the data stream. The solid curve depicts the theoretical optimum, and the stepped levels indicate the quantized density function. The portion of each step which lies above the theoretical optimum represents the excess bandwidth that is required by the quantization.

If the number of substreams N is known, it is possible to determine the length of each segment which provides the most efficient use of the available bandwidth. This determination can be carried out recursively by means of a differential analysis, in which the best choice for t_i, the starting point for segment i, is determined if t_{i-1} and t_{i+1} are known. Referring to Figure 6, the bandwidth is at a local extremum when the two shaded areas in the diagram are of equal area, i.e.,

$$[\nu(t_{i-1}) - \nu(t_i)]dt_i = (t_{i+1} - t_i)[-\nu'(t_i)dt_i] \quad . \tag{18}$$

Rearranging the terms indicates that the derivative of the grain density at t_i must be equal to the slope of the dashed line in Figure 6:

$$\nu'(t_i) = \frac{\nu(t_i) - \nu(t_{i-i})}{t_{i+1} - t_i} \quad . \tag{19}$$

From this relation, t_{i+1} can be obtained as a recursion on t_i and t_{i-1}:

$$t_{i+1} - t_i = \frac{\nu(t_i) - \nu(t_{i-1})}{\nu'(t_i)} \quad (20)$$

$$= -(t_i)^2\left(\frac{1}{t_i} - \frac{1}{t_{i-1}}\right) \quad (21)$$

$$= -(t_i)\left(\frac{t_{i-1} - t_i}{t_{i-1}}\right) \quad (22)$$

$$t_{i+1}t_{i-1} - t_i t_{i-1} = t_i^2 - t_i t_{i-1} \quad (23)$$

$$\frac{t_{i+1}}{t_i} = \frac{t_i}{t_{i-1}} \quad (24)$$

The bandwidth used by a substream is $(t_{i+1} - t_i)\nu(t_i)B$, which simplifies to $[t_{i+1}/t_i) - 1]B$, a constant according to the recursion.

This analysis is based on the assumption that the data rate for a presentation is uniform. In practice, however, compressed data rates can vary in dependence upon the content of the presentation, wherein relatively still scenes may require significantly less than 3Mb/s, whereas action-packed scenes might require more than 6Mb/s. Therefore, it is more useful to express the foregoing relationship in terms of the fragments of a segment rather than the starting time of the segment.

The first fragment of a segment i is represented as n_i, and the segment i contains all of the fragments in the range n_i to $n_{i+1} - 1$. The fragment size G is the amount of memory required to represent one fragment of the presentation. A local extremum of the required bandwidth for a segment i can be expressed as the following induction rule:

$$0 = G\frac{d}{dn_i}[(n_i - n_{i-1})\nu(n_{i-1}) + (n_{i+1} - n_i)\nu(n_i)] \quad (25)$$

$$0 = \nu(n_{i-1}) - \nu(n_i) + (n_{i+1} - n_i)\nu'(n_i) \quad (26)$$

$$\nu'(n_i) = \frac{\nu(n_i) - \nu(n_{i-1})}{n_{i+1} - n_i} \quad (27)$$

The range n_i to n_{i+1}-1 specifies which fragments should be transmitted in substream i. The time $t(n_i)$, the time after pressing "play" at which the lowest-numbered fragment is expected to be available, specifies the maximum length of time to transmit one full copy of segment *i* in its corresponding substream. Although the induction is derived with the values for n_{i-1} and n_{i+1} fixed and with n_i as a variable, the induction can be used to compute n_{i+1} from n_{i-1} and n_i. This provides a way to compute n_i inductively in ascending order, with only one free variable. In particular, since choosing τ fixes the curve $\nu(n)$, and by definition $n_0=0$, the number of fragments in the zero'th substream, n_1, can be regarded as the free variable. Given its choice, n_2 can be computed from n_0 and n_1, and so on.

One approach to choosing an appropriate encoding would be to iterate over all possible values of n_1, applying the induction recursive for each, and choosing the one that wastes the least bandwidth. However, this solution may not be preferable because the high-numbered values of n_i are extremely sensitive to the choice of n_1. Iterating over values of n_1 could miss near-optimal solutions, especially since in the vast majority of candidate solutions generated in that manner, n_N is not likely to match the length of the presentation.

A relaxation procedure is more preferably used to approximate the solution given above. First, locally optimal values are computed for the odd-numbered n_i's, given the even-numbered values. Then locally optimal values for the even-numbered n_i's are computed, given the odd-numbered values. This procedure is iteratively repeated until the bandwidth stops improving.

Playback

At the subscriber's premises, the multiplexed substreams are received and the fragments are reassembled in sequential order, to display the presentation. One embodiment of a decoder 14 for reassembling the encoded data is illustrated in Figure 7. The received data is first presented to a gate 20, which determines

whether a copy of each fragment has been received since the user entered a request to view the presentation. If a given fragment has not been received since the subscriber's request, it is forwarded to a buffer 22. If, however, a copy of the fragment has been previously received and stored in the buffer, all subsequent copies of that fragment are discarded.

In order for the incoming signal stream to be correctly decoded, the gate must be able to identify each fragment that is being received, regardless of when the subscriber enters the request to view the presentation. In one embodiment of the invention, an identifier is included with each transmitted copy of a fragment. Figure 8 illustrates one possible format of a data packet for the transmission of fragments. To minimize bandwidth requirements, the presentation data is compressed, for example in accordance with the MPEG standard. The data packet for a fragment therefore begins with a header 24 which conforms with the MPEG standard. Following the header, a fragment identifier 26 is transmitted. In a straightforward implementation, the identifier could be the integer index n which indicates the location of the fragment in the total data sequence. In other words, the first fragment in the presentation has an identifier of 0, the next fragment is identified as 1, etc. Alternatively, the identifier could be a code value which the gate 20 uses to look up the index number, by means of a previously downloaded look-up table. This latter implementation might be preferable as a mechanism for restricting access to authorized subscribers.

In addition to the index number or code value for the specific fragment being transmitted, the identification data could also include a program identification, or PID, which indicates the presentation to which that fragment belongs. In many cases, the PID for a presentation is incorporated into the MPEG header information. In the event that it is not, however, it can be included with the identifier portion 26 of the transmitted data. Following the identifier 26, the actual data 28 for the fragment is transmitted. In one embodiment of the

invention, the total length of the data packet, including the header and identification data, is 188 bytes.

At the receiver, the gate 20 first examines the PID to determine whether a received fragment belongs to the presentation that has been requested. If the PID is correct, the gate examines the fragment identifier and determines whether that fragment has been previously stored in the buffer 22. For example, the gate might have an associated bitmap 30, in which sequential bit positions respectively correspond to the fragments of the presentation. When a subscriber enters a request to view a presentation, the bitmap is reset so that all bits contain the same value, e.g., zero. As each fragment is received at the gate 20, the bitmap is checked to determine whether the bit corresponding to that fragment's index number has been set. If not, i.e., the value is still zero, the fragment data is forwarded to the buffer 22, for storage, and the bit is set in the bitmap 30. Once the bit has been set, all subsequent copies of that fragment are discarded by the gate 20.

Thereafter, at a time no later than τ seconds after the user has entered a request to view the presentation, a read circuit 23 retrieves the fragments stored in the buffer 22 in sequential order, and successively feeds them to the frame buffer 16 at the proper video rates. The encoding of the substreams ensures that at least one copy of each fragment will have been received by the time that it is required for presentation to the frame buffer 16. Thus, with a maximum wait time of τ seconds, the subscriber is able to view the complete presentation, from start to finish.

In practice, the first copy of each fragment that is received after the subscriber enters a request to view the presentation is stored in the buffer 22. Hence, for at least a short period immediately after the request is entered, every received fragment in each substream is going to be stored. Thereafter, as copies of the fragments begin to be repeated, the received fragments are selectively

discarded. For instance, in the exemplary encoding depicted in preceding Table 1, every incoming fragment is stored for the first nine time slots. Thereafter, as segments begin to repeat, the rate of storage into the buffer 22 begins to decrease.

In the embodiment of Figure 7, the buffer 22 must be capable of writing the data at the rate provided by the gate 20. Hence, it must be capable of operating at the maximum bit rate for transmission. It may not be desirable, or economically feasible, to utilize memory which is capable of operating at such a rate, and which also has sufficient capacity to store all of the fragments that will be needed. An alternative embodiment of the decoder, which permits a slower form of storage to be employed, such as a magnetic hard disk, is illustrated in Figure 9. In this embodiment, a gate 34 receives each incoming fragment and selectively determines whether it is to be kept or discarded. Fragments which are to be kept are forwarded to a fast FIFO buffer 36, which is capable of writing data at the rate provided by the gate. This data is then forwarded to a main buffer 38 at a slower rate, for storage until needed. In this embodiment, the main buffer 38 is sufficiently large to store all of the necessary fragments, up to the entire length of the presentation, and the fast FIFO buffer 36 can be much smaller. In an alternate embodiment, fragments are discarded after being viewed. In this case, the main buffer 38 only needs to be large enough to store a limited portion of the presentation at one time, e.g. 44% of the fragments.

As a further feature of this embodiment, the gate 34 can make a determination whether fragments that are to be kept will spend a negligible length of time in the buffers. For instance, immediately after the subscriber enters a request to view the presentation, the lowest numbered fragments will be needed right away, to begin the playback of the presentation. Later in the presentation, other fragments may arrive just in time for display as well. In this case, therefore, the gate can forward these fragments directly to the read/assembly circuit 40, so that they are immediately available for display. To do so, the gate might be provided with a running clock (not shown) that indicates the number of the current

fragment that is being displayed. If an incoming fragment is within a certain range of that number, it is directly forwarded to the read/assembly circuit 40.

In this mode of operation, the read circuit may receive fragments out of their sequential order. To accommodate this situation, a pre-buffer 42 is located between the read/assembly circuit and the frame buffer 16. This pre-buffer provides random access writing capability, and sequential readout to the frame buffer 16.

To utilize the broadcast-video-on-demand capability provided by the present invention, it is necessary, therefore, that there be sufficient local storage capacity at the subscriber's premises to store and reassemble the fragments from the encoded transmissions. In some cases, however, viewers who do not have such storage capacity may also desire to see the presentation in a non-demand, or conventional, manner. To accommodate this situation, as well as provide additional advantages, discussed below, the bandwidth allocated to the presentation is increased by the amount necessary to transmit the presentation in a conventional manner, e.g., in a sequential form at the nominal rate of 3Mb/s. In the conventional transmission, therefore, the fragments are not transmitted at different rates, so that every fragment has the same repetition period, which is approximately equal to T.

For the exemplary encoding depicted in the foregoing table, in which the bandwidth multiplier $\eta=7.3$, the addition of the conventional broadcast increases the multiplier to a value of $\eta=8.3$. In other words, a total bandwidth of 25Mb/s would be allocated to the presentation. It is to be noted that, using known QAM-64 techniques, it is possible to transmit data at a rate of approximately 27Mb/s within a 6 MHZ channel band. Thus, both a conventional transmission of a 2-hour movie and an encoded transmission with a maximum wait time of 15 seconds can be broadcast within the bandwidth of a single television channel. Throughout the following discussion, embodiments of the invention in which a conventional, sequentially ordered copy of a presentation is transmitted with the

encoded substreams are identified as "layered" transmissions, i.e. one layer of substreams consists of the encoded data, and another layer comprises the sequential data.

Periodic Transmission

In the transmission of the reordered data fragments, it is preferable that the encoding of the presentation be periodic. In other words, the data in all of the substreams should repeat at regular intervals. If the encoding meets this condition, the presentation source 10 only needs to store one period of the encoding. Otherwise, it will be necessary to either store, or generate in real time, an encoding that is long enough to provide continuous service.

Furthermore, it is preferable to have the period of the encoding be related to the length T of the presentation. In such a case, where a conventional layer of the presentation accompanies the encoded layer, as described previously, they can be transmitted together with the same periodicity. This is accomplished by modifying each encoded substream to have a period which is approximately equal to T. To illustrate the manner in which this can be accomplished, a section of one substream which contains enough copies of a segment to span the period T is considered. If the length of that section exceeds the period T, a portion of one segment is deleted from the substream. If the conventionally transmitted version of the presentation does not accompany the encoded version, the deletion of a portion of one segment would violate the condition set forth in relationship (2), since the density of the fragments in the deleted portion would be insufficient. For instance, Figure 10a illustrates a substream that complies with the conditions of relationship (2). In this substream, a given fragment, indicated by the shaded area, periodically appears in the substream. The maximum separation between successive copies of the fragment complies with the density requirements of relationship (2). Figure 10b illustrates a modified substream, in which a portion of a segment, which includes the given fragment, has been deleted. As can be seen,

the distance between successive copies of the given fragment exceeds the maximum separation permitted by relationship (2).

However, it is possible to delete a portion of a segment if, for each deleted fragment, a copy of that same fragment is present at an appropriate location in another substream. In the context of this disclosure, a copy of a fragment that is present in another substream is called a "proxy". In a practical implementation of the invention, the conventionally transmitted presentation can be used to provide proxies for deleted portions of encoded substreams. Referring to Figure 10c, when a proxy is present, it is possible to delete fragments if, for each deleted fragment, a proxy begins no earlier than one maximum separation period before the end of the next fragment copy in the substream, and ends no later than one maximum separation after the end of the previous copy of the fragment. This requirement ensures that the fragment, or its proxy, fulfills the density conditions of relationship (2). The possible locations of a proxy, for the deleted fragment of Figure 10b, are illustrated in Figure 10c. As long as a copy of the deleted fragment is present in the proxy stream at one of these locations, it can be deleted from the original substream, since one copy will still be received at the subscriber's premises by the required time.

When multiple contiguous fragments are deleted within a segment, this condition can be characterized more generally. If the number of fragments to be deleted from a substream i is denoted as d_i, then a copy of fragment j can be deleted from substream i only if a proxy exists in another substream with an offset Δ that lies in the range $0 \leq \Delta < d_i$. A proxy offset is defined as the column number, or transmission time slot, of the fragment copy to be deleted, minus the column number of the copy which serves as a proxy.

Through the use of proxies in this manner, each substream can be modified to provide it with periodicity T. Figure 11 illustrates an example of encoded substreams that are transmitted together with a conventional stream. To modify the encoded substreams to provide periodicity T requires that up to one full

segment, minus one fragment, may need to be deleted from each encoded substream. If the fragments are deleted after the end of the period T, as indicated by the shading, the requirements of relationship (2) may be violated. However, the encoded substreams can be time shifted to permit deletion of a portion of a segment, and utilize proxies from the conventional stream. Referring to Figure 12, the fragments are arranged in ascending sequential order within each segment of an encoded substream. In the substream, a target range of five fragments, identified as A-E, is to be deleted, so that $d_i=5$. The substream is time shifted so that a copy of the first fragment in the target range, i.e., fragment A, is aligned with a copy of any fragment in the conventional stream that is within the target range. In this particular example, the conventional stream is time-division multiplexed over three substreams.

Regardless of how the encoded substream is time shifted relative to the conventional stream, the first fragment always has the smallest proxy offset, and the last fragment always has the largest proxy offset. Since the first fragment's proxy offset cannot be negative, and the last fragment's proxy offset cannot exceed d_i-1, every proxy offset is within the required range. In the illustrated example, fragment A has an offset of one, and fragment E has an offset of 3, which meets the requirements for removal of fragments. Thus, through appropriate deletion of portions of segments, and time shifting of the conventional stream and the substreams relative to one another to properly align the proxies of the deleted fragments, it is possible to ensure that each encoded substream has a periodicity T.

In a case of a layered transmission, the periodicity of each substream can be made approximately equal to the total running time T of the presentation. If layering is not employed, the periodicity of the encoded substreams can be made smaller than T. In this case, however, one or more additional substreams, which function as proxy substreams, need to be transmitted. Each encoded substream can be modified to have a desired periodicity, R, if a number of fragments equal to $\gamma(t(n) \bmod R)$ are deleted from the substream. The length of the repeating portion

of the substream is equal to t(n), and γ is equal to the number of fragments that can be transmitted per unit time at bandwidth $\lambda\eta$. For a given R, therefore, the proxy substream must contain P fragments, where $P=\gamma\Sigma_n$ (t(n) mod R). Since γ(t(n) mod R) falls within the range (0, γt(n)), the value for P therefore lies in the range (0, γT-N). The proxy fragments can therefore be transmitted in a number of proxy substreams equal to the smallest integer at least as large as P/R. In most cases, the proxy substreams will not be full, which permits the required bandwidth to decrease when no proxy fragments are needed. It is possible to employ proxies for up to half of the encoded fragments. As a result, encodings with periodicities which are considerably less than 0.5T can be accomplished.

Switching of Presentations

As long as an encoded presentation is continuously transmitted in such a periodic fashion, a subscriber can view or otherwise utilize a presentation at any arbitrary time, with a maximum wait time of τ. In a practical implementation, however, the providers of broadcast services will need to terminate the transmission of an encoded presentation, to change to a new presentation. For instance, in a subscription television system, movies may be changed on a daily or weekly basis. Consequently, at a given point the encoded transmission of one movie will terminate, and the transmissions of the next movie will commence. This situation is schematically illustrated in Figure 13, for an example where each presentation is transmitted over four substreams. Vertical lines within a substream indicate the boundaries between repeated segments. For illustrative purposes, the segments are illustrated so that their boundaries align at the time S when the switch is made from one presentation to the other.

When this switch occurs, there exists a period of time during which a subscriber cannot enter a request to view the first movie and be able to receive the entirety of that movie. This period begins when the last copy of any fragment in the first presentation is received at the subscriber's premises. If the subscriber

presses the "play" button any time after the point at which it is possible to capture that copy of the fragment in its entirety, it is not possible to view the complete presentation. As a practical matter, this period of time has a duration which is equal to the length of the longest segment. In a further aspect of the invention, this period of time is minimized, to thereby reduce the length of time that a subscriber is prohibited from the viewing the first movie in its entirety and must wait for the start of the second movie.

The last point in time at which the user can press the "play" button and still view the presentation in its entirety, before the switch to a new presentation, is designated as the point L. Beginning at this point, it is only necessary to receive one copy of each fragment, to ensure that the entire presentation can be viewed in response to any actuation of the "play" button up to the point L. Any additional copies of fragments that are transmitted after this time will not be used, and are therefore unnecessary. Figure 14 illustrates the transmission of one copy of each fragment, i.e. one complete segment, in each substream, beginning at time L. As indicated by the blank areas, there is unused capacity in all but the highest-numbered substream. If each segment continues to be repetitively transmitted, this unused capacity is occupied by redundant copies of presentation segments. One way to reduce the duration of the period between L and S, therefore, is to move some of the fragments from the higher-numbered substreams into the available bandwidth of lower-numbered substreams, so that exactly one copy of each fragment is transmitted after the point L, using the full available bandwidth, as depicted in Figure 15.

In so doing, however, attention must be paid to the order in which the fragments are transmitted, to ensure that the fragment density requirements of relationship (2) are maintained. A straightforward approach that can be employed to ensure this condition is to time-division multiplex the substreams in their usual order, but to omit redundant copies of fragments. This approach guarantees that every fragment will be at least as close to the last start time L as it was before the

redundant fragments were deleted. The transmission of the last fragment n_{max} marks the point S at which the transmission of the encoded substreams for the new presentation can begin. In an un-layered embodiment of the invention, the reordering of the fragments in this manner provides a time period S-L that has a duration of $T/\lambda N$. In the case of the sample encoding depicted in Table 1 for a two hour movie, this results in a period of about 16.5 minutes during which the subscriber cannot begin the playback of either movie.

In the case of a layered embodiment, the additional bandwidth provided by the conventional layer permits this period of time to be reduced even more. Two alternative approaches are possible, as respectively illustrated in Figures 16 and 17. In these figures, the conventional transmission is depicted by the horizontal layers on top of the encoded substreams. Figure 16 illustrates the situation in which the final transmission of each fragment, represented by the shaded area, occurs after the conventional transmission has terminated. In this case, the bandwidth normally allocated to the conventional layer can also be used for the encoded transmission. In other words, the encoded transmission has a total available bandwidth of $(\lambda N+1)\eta$. Consequently, the period S-L is reduced to a duration of $T/(\lambda N+1)$. In the case of the sample encoding, therefore, this results in a non-viewing period of about 14.5 minutes for a two hour movie.

Of course, the implementation of Figure 16 means that the conventional viewing of the movie also has the same blackout period. In the alternative implementation of Figure 17, the bandwidth allocated to the last copy of each fragment is reduced, so that the conventional presentations can abut one another. This approach would appear to have the same effect as the non-layered approach of Figure 15, and lengthen the period S-L to $T/\lambda N$ for the encoded version of the presentation. However, using the conventional stream as a proxy, a sufficient number of fragments can be removed from the encoded portion of the transmission to provide a non-viewing period of duration $T/(\lambda N+1)$. In essence, all of the fragments in the last $T/(\lambda N+1)$ seconds of the presentation are available from the

conventional layer, and are received at least as early as they would be in the highest-numbered substream. Consequently, they can be omitted from the encoded substreams, thereby shortening the period of time necessary to transmit one copy of every fragment.

In each of the foregoing embodiments, the boundary between the first presentation and the second presentation occurs at the same time S for all of the substreams. In an alternative implementation, different switching times are employed for each substream. Referring again to Figure 14, it can be seen that the bandwidth requirements for the first presentation are reduced as the transmission of each segment is completed. These reduced bandwidth requirements can be used to simultaneously ramp up the bandwidth that becomes available for the second presentation. To illustrate, Figure 18 depicts the time relationship of the substreams for an encoded transmission. In this diagram, time t=0 is the moment at which the subscriber presses the "play" button. As soon as this event occurs, copies of the fragments are captured from all of the various substreams. In particular, the capturing operation does not need to wait until the first fragment of the substream appears on its designated substream. The fragments are retrieved in the order in which they appear, beginning at time 0. Furthermore, the playback does not begin as soon as the first fragment has been received. Rather, to guarantee uninterrupted display of the presentation, the playback of a segment *i* begins after one full copy of that segment has been transmitted over the time period $\tau(1+\lambda)^i$ from time 0. Thus, the playback of the first segment begins when the time period τ has elapsed, at time t_0, once one copy of each fragment in that segment has been received.

For any given segment *i*, a full copy of that segment is received between time 0 and time t_i, and plays from time t_i to time t_{i+1}, which defines an interval of duration λt_i. However, each segment is transmitted over the interval of time $\tau(1+\lambda)^i$ which is as long as the sum of the viewing times of all preceding

segments, plus the wait time τ. Thus, each segment can be transmitted much more slowly than its own viewing time.

In the layered implementation, the first required copy of each fragment appears in the conventional layer. Consequently, the transmission of an encoded segment i need not begin until time t_{i+1} in order to meet the conditions of relationship (2). As a result, the substreams of the second presentation can start with an irregular boundary that complements the irregular termination boundary of the first presentation, as depicted in Figure 19, with sufficient room between them to avoid problems. Specifically, substream i of the first presentation must play until time t_i, and substream i of the second presentation need not start until time t_{i+1}. If the two presentations have identical encodings, e.g., they are both movies of the same length, then the inter-substream gap between the end of the first presentation and the beginning of the second presentation is λt_i. Otherwise, the gap is $(t_{i+1})-t_i$, where t_{i+1} is determined relative to the encoding of the second presentation, and t_i is determined with respect to the encoding of the first presentation.

An exemplary encoding for a movie which incorporates the foregoing concepts will now be described. As in the example of Table 1, a two-hour movie is considered. The movie is divided into fragments of 188 bytes each, resulting in approximately 14.4 million fragments. The value for λ is chosen to be 1/25, and the bandwidth multiplier is approximately equal to 8. Consequently, 25 substreams are allocated to the conventional transmission of the movie, and the number of encoded substreams $N=175$. Pursuant to Equation 8, the wait time τ is a function of the movie length, and is approximately 7.5 seconds. Given these parameters, therefore, the first two segments of the movie contain 602 and 627 fragments, respectively.

The encoding of the movie is created by time-division multiplexing the 175 encoded substreams with the $1/\lambda=25$ conventional substreams. As in the example of Table 1, the individual substreams are represented by the rows of the table, and

each column represents one transmission slot. For the conventional transmission, the fragments of the presentation are allocated to the 25 substreams in raster order, as depicted in Table 2 below.

Table 2

Substream	Fragment Sequence										
0	0	25	50	75	100	125	150	175	200	225	•••
1	1	26	51	76	101	126	151	176	201	226	•••
2	2	27	52	77	102	127	152	177	202	227	•••
3	3	28	53	78	103	128	153	178	203	228	•••
4	4	29	54	79	104	129	154	179	204	229	•••
5	5	30	55	80	105	130	155	180	205	230	•••
•	•	•	•	•	•	•	•	•	•	•	
•	•	•	•	•	•	•	•	•	•	•	
•	•	•	•	•	•	•	•	•	•	•	

The encoded substreams begin at Substream 25 in this example. The assignment of the fragments to the encoded substreams is carried out with reference to the fragments in the conventional substreams. The first encoded substream contains fragments 0 through 601. To determine its appropriate location, fragment 601 is identified in the conventional substream. Beginning in the next column, i.e. transmission slot, of the encoded substream, fragments 0-601 are sequentially inserted, and this sequence is repeated until one segment length beyond the end of the conventional data stream. This encoding is depicted in Table 3 below.

Table 3

Fragment Sequence																		
Conventional substreams 0-24	500	525	550	575	600	625	650	675	700	725	•••	15,600	15,625	#####	15,675	#####	#####	•••
	501	526	551	576	601	626	651	676	701	726	•••	15,601	15,626	#####	15,676	#####	#####	•••
	502	527	552	577	620	627	652	677	702	727	•••	15,602	15,627	#####	15,677	#####	#####	•••
	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•••
	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•••
	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•••
	523	548	573	598	623	648	673	698	723	748	•••	15,623	15,648	#####	15,698	#####	#####	•••
	524	549	574	599	624	649	674	699	724	749	•••	15,624	15,649	#####	15,699	#####	#####	•••
Substream 25						0	1	2	3	4	•••	599	600	601	0	1	2	ooo

The foregoing examples are presented for the case in which the bandwidth requirements are uniform, such that the temporal spacing of fragments is the same for all segments of the presentation. As noted previously, however, the data rates, and hence the bandwidth requirements, can vary in dependence upon the content of the presentation. As a result, one segment of the presentation may require a significantly greater number of fragments to be transmitted in a given time than another segment of the movie. When the fragments of these two segments are multiplexed, consideration must be given to these differing data rates.

In accordance with another aspect of the present invention, a technique identified as periodic-queue time-division multiplexing is employed to accommodate variable rate data streams. Each substream comprises a series of events, where each event is the beginning of the transmission of a fragment. Each event is associated with a nominal time, and within a given substream the nominal times occur at regular intervals. The length of that interval, which is therefore equal to the temporal length of a fragment, can be expressed as follows:

$$\Delta t_i = \frac{t(n_i)}{n_{i+1} - n_i} \tag{28}$$

where: $t(n_i)$ is the temporal length of segment i, and

$n_{i+1} - n_i$ is the number of fragments in segment i.

The value of Δt_i can be different for different values of i, i.e. different substreams, in the case of variable bandwidths.

The objective of periodic-queue time-division multiplexing is to assign each event an actual transmission time such that, when the events from all of the substreams are multiplexed together, the actual times occur at regular intervals. This is accomplished by assigning each fragment a nominal broadcast time. Figure 20 illustrates the nominal times that are assigned to the fragments in three segments having different nominal time intervals Δt_i. Once the nominal times have been assigned, the fragments are sorted in order of nominal time. In the case

of ties, the fragments with the same nominal time can be arbitrarily ordered, e.g., in accordance with their segment number. The fragments are then transmitted in the sorted order at equally spaced actual times, to provide a fixed data rate. The interval Δt_{mux} between the actual times is chosen as the harmonic mean of the nominal time intervals:

$$\frac{1}{\Delta t_{mux}} = \sum_i \frac{i}{\Delta t_i} \tag{29}$$

As a result, each of the fragments is broadcast, on average, at the appropriate frequency. Furthermore, the difference in the nominal-time interval between any two events, e.g. successive copies of the same fragment, and the actual-time interval between those same events is bounded. For any two events whose nominal times are separated by a duration t, the number of intervening events that occur within each substream k is an integer $(t/\Delta t_k) + \delta_k$, where δ_k is a real-valued number in the range (-1, +1). The total number of intervening events from the entire data stream is therefore $(t/\Delta t_{mux}) + \Sigma_k \delta_k$. Consequently, the actual-time difference between two events differs from the nominal time difference, t, by the quantity $\Delta t_{mux} \Sigma_k \delta_k$. This quantity lies within the range $(-N\Delta t_{mux}, +N\Delta t_{mux})$. In other words, for any two events, the difference between the actual and nominal times is bounded by N units of Δt_{mux}. Hence, the cost associated with the ability to provide a constant bandwidth data stream is an increase of the actual wait time, beyond the nominal wait time τ, by a quantity $(N-1)\Delta t_{mux}$, where $\Delta t_{mux} = G/\eta m$. For the example described in connection with Tables 2 and 3, this amounts to an increase of about 12.5 ms.

When a program switch occurs, the number of substreams N might change if the two programs are not of the same length. Further, if the embodiment of Figure 19 is employed, the various substreams stop and start at staggered times. The periodic-queue time-division multiplexing technique should be applied in a manner which takes these factors into consideration. To do so, a grid such as

those of Tables 2 and 3 is defined, having an infinite number of columns and a number of rows equal to the maximum anticipated value of N. The fragments of the presentations are then scan-converted into the grid in order of nominal times, using only those cells which correspond to substreams that are currently active. Referring to Figure 19, for example, during the period from time 0 to t_0, no fragments are placed in the rows of the grid which pertain to the conventional substreams, i.e. rows 0-24 in the example of Table 3. At time t_0, the fragments of the second conventional presentation begin to be loaded. At this same time, encoded fragments from the first presentation are no longer loaded into the next row, e.g. row 25, although they continue to be loaded into all subsequent rows. At time t_1, the fragments of the second presentation begin to be loaded into row 25, and fragments of the first presentation are no longer loaded into row 26. The process continues in this manner, until a complete switch from the first to the second presentation has taken place.

From the foregoing, therefore, it can be seen that the present invention provides the ability to broadcast temporally ordered data in a manner which enables a user to utilize that data with a minimal waiting period, regardless of the point in time at which the request to utilize the data is made. In its application to the transmission of video presentations, a user is able to begin viewing a two hour movie with a maximum wait time of approximately 7.5 seconds. Using currently available transmission techniques, these results can be accomplished within the bandwidth that is allocated to a conventional television channel, while at the same time permitting a conventional transmission of the movie to take place as well. Hence, every available television channel is capable of providing a different movie in both a conventional mode and a video-on-demand mode. The number of different movies that can be viewed in the video-on-demand mode is not limited by the storage capacities at the viewer's site. Rather, it is determined only by the number of channels that the television service provider allocates to the movies.

The efficiency of the present invention can be readily ascertained by reference to two dimensionless parameters, bandwidth ratio and time ratio. The bandwidth ratio η is a factor by which the bandwidth of an encoded transmission exceeds that of a conventional broadcast of the same presentation. This is a measure of the cost to the broadcaster. The time ratio T/τ is the factor by which the presentation length T exceeds the wait time τ, and is therefore a measure of the benefit to the viewer. The graph of Figure 21 illustrates the relationship of these two ratios for three cases, near-video-on-demand (curve 42), pyramid broadcasting as described in the previously cited references (curve 44), and the present invention (curve 46). As can be seen, for near-video-on-demand the time ratio is linear in the bandwidth ratio. Pyramid broadcasting provides improved results. For example, a bandwidth ratio of 10 has a time ratio of 62, and a bandwidth ratio of 16 has a time ratio of 248. In the present invention, a bandwidth ratio in the range of 6-8 provides a time ratio of 400-3000, e.g. two-to-twelve second wait times for a two-hour presentation.

The features of the present invention can be combined with other known techniques to provide additional enhancements. For instance, the previously described approach, in which the beginning portions of movies are stored ahead of time at the subscriber's premises for instant retrieval, can be employed in conjunction with the invention to eliminate the wait time τ altogether. In this case, however, only a very small portion of the movie needs to be preliminarily stored at the subscriber's premises, e.g. 7.5 seconds for a two-hour movie, thereby significantly reducing the local storage requirements per movie. As an alternative, material which is common to several presentations, such as a copyright warning, a rating announcement, a studio logo, or an advertisement, can be locally stored and played during the initial τ seconds, to eliminate the perception of any waiting time.

It will be appreciated by those of ordinary skill in the art that the present invention can be embodied in other specific forms without departing from the spirit or essential characteristics thereof. For instance, the foregoing examples of the invention have been described for the cases in which each segment contains the maximum integral number of fragments that fit within the repetition period of the segment. In this case, in order to ensure that each fragment arrives at the subscriber's premises within the required time, the fragments are transmitted in the same sequence within each fragment. However, if it is desirable to be able to reorder the fragments from one segment to the next, it is possible to transmit less than the maximum number of fragments within the period of a segment. In this case, the extra space that is available within the segment can be used to reorder the fragments, e.g. transmit a given fragment before it is required, while still preserving the time guarantees.

Similarly, while the use of equal-size fragments has been described in the foregoing examples, particularly to facilitate the multiplexing of the substreams, other embodiments of the invention might employ fragments of varying sizes, for instance where the substreams are transmitted independently of one another, in parallel.

Furthermore, while the foregoing embodiments of the invention have been specifically described with reference to their application to televised movies, it will be appreciated that the principles which underlie the invention can be applied to any type of temporally-ordered data that is distributed via a broadcast or multicast medium. The presently disclosed embodiments are therefore considered in all respects to be illustrative, and not restrictive. The scope of the invention is indicated by the appended claims, rather than the foregoing description, and all changes that come within the meaning and range of equivalence thereof are intended to be embraced therein.